Available online at www.sciencedirect.com
ScienceDirect
Computer Speech and Language 25 (2011) 386–403
COMPUTER SPEECH AND LANGUAGE
www.elsevier.com/locate/csl

Document sentiment classification by exploring description model of topical terms

Yi Hu [a,b,*], Wenjie Li [b]

[a] *Department of Search Platform, Tencent Communications Corp., Shenzhen, China*
[b] *Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China*

Received 5 January 2009; received in revised form 6 July 2010; accepted 15 July 2010
Available online 23 July 2010

Abstract

Sentiment classification is used to identify whether the opinion expressed in a document is positive or negative. In this paper, we present an approach to do documentary-level sentiment classification by modeling description of topical terms. The motivation of this work stems from the observation that the global document classification will benefit greatly by examining the way of a topical term to give opinion in its local sentence context. Two sentence-level sentiment description models, namely positive and negative Topical Term Description Models, are constructed for each topical term. When analyzing a document, the Topical Term Description Models generate divergence to support the classification of its sentiment at the sentence-level which in turn can be used to decide the whole document classification collectively. The results of the experiments prove that our proposed method is effective. It is also shown that our results are comparable to the state-of-art results on a publicly available movie review corpus and a Chinese digital product review corpus. This is quite encouraging to us and motivates us to have further investigation on the development of a more effective topical term related description model in the future.

Keywords: Sentiment classification; Topical term; Topical Term Description Model; Maximum spanning tree

1. Introduction

Sentiment classification is a recent rapidly growing sub-discipline of text classification concerned with opinion expressed in a text rather than its topic. Document-level sentiment classification is aimed at classifying a document according to the positive or negative polarity of its opinion. Automatically labeling review documents such as product reviews or movie reviews with their sentiment polarities can be useful in many business intelligent systems and recommending systems.

Conventional topic-oriented classification models normally represent a document as a set of terms in which topic sensitive words are important. In contrast, polar terms such as “excellent” and “worst” are considered essential to sentiment-oriented classification. However, we argue that sentiment structures in sentence context are more expressive than individual polar term based features. Take the following sentence selected from a movie review as an example.

* Corresponding author. Tel.: +86 755 8601 3388 84543.
E-mail addresses: sunracerhu@tencent.com (Y. Hu), cswjli@comp.polyu.edu.hk (W. Li).

0885-2308/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.csl.2010.07.004

Taken with fargo, the writer's last product, it is true that the **film satisfies** all **audiences**.

Clearly, the key polar term "satisfies" in the context of "film" is a strong clue to its positive opinion orientation towards the subject matter, i.e. to praise the film. Now let us look at another example.

. . . and the **film never satisfies anyone** beyond its visuals . . .

Although "satisfies" gives a positive perspective in the first example, the negation word "never" in the second example, however, transforms it into a negative one. This implies that the same polar term may deliver different perspectives when it appears in different contexts. The polarity transformation issue has been addressed by defining negation rules. For example, Kennedy and Inkpen (2006) take into account contextual valence shifters, such as negations and intensifiers in addition to counting the positive and the negative terms. In their study, negations are used to reverse the semantic polarity of a particular term, while intensifiers are used to change the degree to which a term is positive or negative. Hand-crafted rules encode human knowledge and therefore are of high accuracy in narrow domains. However, they are not always robust in natural language processing applications. In our consideration, more helpful evidence should come from a broader, associated and structured context in the second sentence, such as "never satisfy anyone", that can be represented by two pairs, i.e. $< never, satisfy >$ and $<satisfy, anyone>$. Note that when the object of "satisfy" (i.e. "anyone") is replaced with "everyone", the polarity of the whole phrase "never satisfy everyone" becomes neutral. Such polarity transformation could not be achieved by rules alone, but the pair $< satisfy, everyone >$ might provide the necessary polar information. In order to better capture sentiment information, we propose an approach to mine the content structures of topical terms in sentence-level contexts, and accordingly develop the named Topical Term Description Model (**TTDM** for short) for sentiment classification.

In our definition, "topical terms" are those specified entities or certain aspects of entities in a particular domain, such as "film" and "actor" in movie reviews. Because it is difficult to pre-define a complete set of topical terms for any given domain, we introduce the automatic extraction of topical terms from text based on their domain termhood and use these extracted terms to approximately characterize document topics. A sentence containing at least one topical term is called a "sensitive sentence" in this paper, which is supposed to express some opinion about the topical term. We assume that the sentiment orientation of a sensitive sentence is hidden in its content context, and the content attached to each topical term can be represented by some structure within the specified context surrounding the topical term. This kind of structure is supposed to convey sentiment information. When identifying the polarity of a sensitive sentence, the generation probabilities of its content structure regarding the positive or negative polarities are considered for classification. In this connection, we need to train two TTDMs for each topical term in order to support probability calculation of such structures. The description models manage to capture the habitual use of a language within the context of a topical term. In this study, we explore the maximum spanning tree (MST for short) where the root node is a topical term in a sensitive sentence. Based on the TTDMs built on MSTs, we can then predict orientations of the sensitive sentences in any new document. These sentences together decide the overall orientation of the document.

The rest of this paper is organized as follows. Section 2 introduces the proposed TTDM in detail and Section 3 presents the model parameter estimations and the smoothing techniques. Section 4 conducts evaluations. Section 5 briefly reviews related work. Finally, Section 6 concludes the paper.

2. Topical Term Description Model (TTDM)

We introduce a topical term description modeling approach to document sentiment classification in this section. The models are developed to capture the sentiment among a topical term and its context from both the positive and the negative perspectives. The motivations of this approach are two-fold.

1. First, the context helps to determine the content structure of a topical term.
2. Second, the generation probabilities of a certain content structure seen from "positive" and "negative" TTDMs are likely to be substantially different.

As mentioned in Section 1, when we analyze a document, the TTDMs generate divergence that supports sentence classification that in turn can be used collectively to decide the whole document classification.

2.1. A general probability framework for sentence sentiment classification

In this paper, we assume that the combination of a topical term and its content structure in a sentence holds sentiment information. Therefore, the content structure (Λ) of a topical term (t) in a sensitive sentence (s) can be formulated by some general probabilistic framework. We choose the following log-ratio decision function f to determine the polarity of a sensitive sentence s.

$$\begin{aligned} f(s) &= log\left(\frac{Pr(POS|s)}{Pr(NEG|s)}\right) = log\left(\frac{Pr(POS|t,\ \Lambda)}{Pr(NEG|t,\ \Lambda)}\right) \\ &= log\left(\frac{\left(\frac{Pr(POS|t)\,Pr(\Lambda|POS,\ t)}{Pr(\Lambda|t)}\right)}{\left(\frac{Pr(NEG|t)\,Pr(\Lambda|NEG,\ t)}{Pr(\Lambda|t)}\right)}\right) = log\left(\frac{Pr(POS|t)\,Pr(POS|t,\ \Lambda)}{Pr(NEG|t)\,Pr(NEG|t,\ \Lambda)}\right) \end{aligned} \tag{1}$$

where s is further extended by a tuple of t and sentiment structure Λ of t, i.e., $s = (t, \Lambda)$. *POS* indicates the positive class tag and *NEG* the negative class tag. In Eq. (1), it is not necessary for t to have its own polarity because the topical terms in this study are objective entities or aspects of entities. Therefore $Pr(POS\,|\,t)$ is considered equal to $Pr(NEG\,|\,t)$. Both of them can be ignored from the function. We have the following approximation.

$$f(s) = log\left(\frac{Pr(\Lambda|POS,\ t)}{Pr(\Lambda|NEG,\ t)}\right) \tag{2}$$

The probabilistic framework illustrated in Eq. (2) allows the generation probabilities of Λ to be used for the sentiment classification of s.

It should be noted that Λ here is a general representation of the sentiment structure. Referring back to the examples in Section 1, the sentiment structure Λ of the context "never satisfy any audience" can be $< never,\ satisfy >$ and $<satisfy,\ audience>$, though the ideal structure might be a higher order "$< never,\ < satisfy,\ audience >>$". The higher order structure is able to capture more accurate information for classification, but it also suffers from more severe data sparseness. In practice, the first order structure might be a compromise when only a small-sized corpus is available. Hence, pair independence is assumed between $< never,\ satisfy >$ and $<satisfy,\ audience>$ at present.

2.2. Maximum Spanning Tree

As just mentioned, the sentiment structure Λ can be represented by any suitable formal structure. In this paper, we exploit the Maximum Spanning Tree (MST) structure to discover the links among the topical term "t" and its context words. For natural language processing applications, the nodes in a MST are usually the focused language units, i.e. words or phrases, while the edges between the nodes indicate their associations.

We now discuss the sentence representation by using MST. For the MST choice, we have the following considerations:

1. It is hard to decide which words in the context of "t" ought to be included or excluded from the sentiment structure. So all the words (except stop words) in the context are taken into consideration. Based on this model, a word should link to at least one other word, and no isolated word is allowed.
2. Calculating all the links between the words is unnecessary, otherwise, useful links for sentiment would be counteracted by useless links. On the other hand, calculating the useless links is time-consuming. So only the most significant links are preserved.

Obviously, the tree style of MST can be used to cover all the non-stop words and has the least number of links (Rigsbergen, 1979). Therefore, when the links between the nodes form a loop, we simply prune the weakest one in the loop in order to guarantee a tree. The MST is expected to capture the most significant sentence-level word links for sentiment classification.

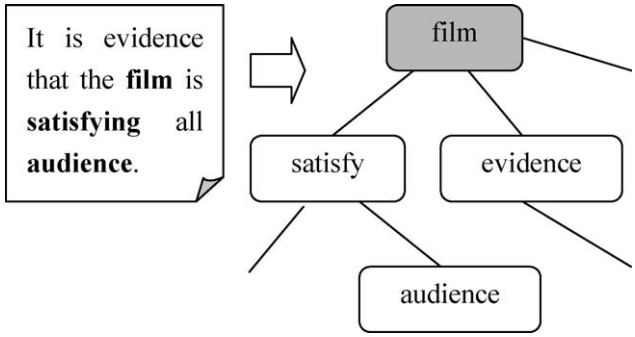

Fig. 1. The partial MST of the instance "It is evidence that the film is satisfying all audience." (The gray node is the root).

According to the assumptions above, a sentence can be easily modeled as a MST structure. The partial MST of the sentence "It is evidence that the **film** is **satisfying** all **audience**" is illustrated in Fig. 1, where the topical term "film is the root node in this MST.

Similarly, the partial MST of the sentence ". . . and the **film never satisfies** any **audience** beyond its visuals . . ." is illustrated in Fig. 2, where the topical term "film" is also defined as the root node in this MST.

2.3. MST construction algorithm

In constructing the MST structure, the weight of the link between a pair (¡satisfy, audience¿ for example) is measured by the Pointwise Mutual Information (PMI) (e.g., Manning and Schutze, 1999; Turney, 2002).

$$weight\,(satisfy, audience) = log\left(\frac{Pr\,(satisfy, audience)}{Pr\,(satisfy)\;Pr\,(audience)}\right) \tag{3}$$

where Pr (satisfy), Pr (audience) and Pr (satisfy, audience) are obtained from all the training sentences (including both sensitive and non-sensitive sentences) which represent the "review" topic. These probabilities are computed by using Maximum Likelihood Estimation. For a sentence with n words, its MST must have $n-1$ edges because of the acyclic characteristic of a tree. The $n-1$ links with the highest weights can link all the n words and build the MST of the sentence. The MST construction algorithm for a sensitive sentence is shown in Fig. 3.

Obviously, this is a greedy algorithm. Step 1 requires computing the PMI for all word pairs in the sensitive sentence, each of which can be done in constant time. Thus, this step has a complexity of $O(n^2)$. "n" is the non-stop word number in this sentence. Then the algorithm ranks all the PMI weights by some sorting method and here we use the quick sort, and the running time of this step is $O(n^2 log(n))$. From Step 5 to Step 10, the algorithm keeps the minimum word pair set that covers all the words in the sentence by choosing decreasing weighted edges. Hence, this step has a complexity of $O(n^2)$. The overall running time for the sentence is thus $O(n^2 log(n))$.

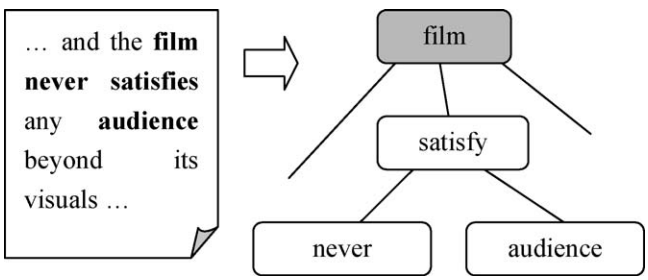

Fig. 2. The partial MST of the instance "and the film never satisfies any audience beyond its visuals" (The gray node is the root).

Algorithm 1: MST Construction Algorithm

Input: A sensitive sentence $s^* = w_1, w_2, \ldots w_n$, where w_t is current topical term.
All stop words have been removed from s^*.

Output: the MST as the solution.

1: for $\forall i, \forall j\ (i \neq j)$, calculate $u_{ij} = \mathrm{PMI}(w_i, w_j)$

2: add all u_{ij} to a weight set U;

3: rank the weight set U by *QuickSort*(U);

4: choose the maximum u_{tk} from all the pairs containing the topical term w_t
to initialize MST = $\{< w_t, w_k >\}$, and remove u_{tk};

5: **while** MST dose not cover all non-stop words in s^* **do**

6: choose the maximum u_{ij} in U

7: if $w_i \notin$ MST or $w_j \notin$ MST

8: add $< w_i, w_j >$ to MST and remove u_{ij} from U;

9: else remove u_{ij} from U;

10: **end while**

11: Output the tree MST with w_t as root ;

Fig. 3. MST construction algorithm.

2.4. Generation probability of MST

We assume that the MST generation probability of the sentence in the positive TTDM is $Pr\,(MST|POS, t)$, and the generation probability in the negative TTDM is $Pr\,(MST|NEG, t)$. If $Pr\,(MST|POSt)$ is larger than $Pr\,(MST|NEG, t)$, the sentence is determined as positive, and negative if otherwise.

It should be noted that a sentence may contain more than one topical term or it may create more than one term-structure (t, Λ) tuple. That means, for a fixed t, the other topical terms in the same sentence are all regarded as ordinary words. Each (t, Λ) from the same sensitive sentence s has its own generation probability, and the largest one decides the polarity of the sentence, i.e., its posterior probability is,

$$Pr(Y|s) = \max_{t \in s} Pr(\Lambda|Y, t) \tag{4}$$

where Y indicates the tag of *POS* or *NEG*. Eq. (4) applies to sentences having more than one topical term. A general MST structure is illustrated in Fig. 4.

In Fig. 4, the topical term "t" is the root of a MST, $w_1 - w_8$ are the context words. The reason for choosing t as the root is that we use the sentence context to express the opinion (implicit or explicit in $w_1 - w_8$) instead of the topical term t. Yet the term t is the trigger of this expression and therefore we start from it to calculate the generation probability of a MST. In the tree structure, each non-leaf word node can link to more than one word node, while the leaf nodes have only one node to link to. To calculate the MST, we split the whole MST into various individual sub-trees. In this paper, a sub-tree is a tree which only comprises a local parent and its directly linked children.

Fig. 5 above illustrates a sub-tree instance from the MST. Taking it as an example, we explain how the sub-tree generation probability given t and Y can be calculated by Eq. (5).

$$Pr\,(subtree|Y, t) = Pr\,(w_1|Y, t) \times Pr\,(w_4|w_1, Y, t) \times Pr\,(w_5|w_1, Y, t) \tag{5}$$

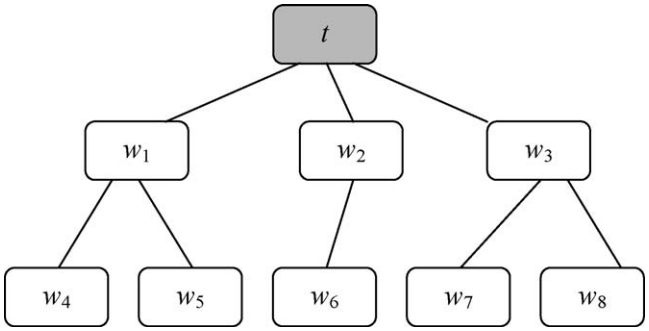

Fig. 4. General Structure of a sentence MST with t as root.

where $Y=POS$ or $Y=NEG$. Eq. (6) calculates the generation probability of the parent node w_1 in one sub-tree. If we assume the independence of sub-trees, we will have the generation probability of the whole MST in Eq. (6).

$$Pr(MST|Y,t) = \prod_{subtree \in MST} Pr(subtree|Y,t) = Pr(t|Y,t) \prod_{w \in W, w \neq t} Pr(w|Y,t) \prod_{<w_i,w_j> \in MST} Pr\left(w_j|w_i,Y,t\right) \quad (6)$$

where W indicates the bag-of-words of s, and the pair $< w_i, w_j >$ indicates a link between w_i and w_j in the MST of s. Because the topical term t is the given condition, $Pr(t|Y,t)$ s equal to "1" and can be ignored.

The polarity of the sentence based on Eq. (6) can then be determined by

$$\begin{aligned} f(s) &= \log\left(\frac{1 \times \prod_{w \in W, w \neq t} Pr(w|POS,t) \times \prod_{<w_i,w_j> \in MST} Pr\left(w_j|w_i,POS,t\right)}{1 \times \prod_{w \in W, w \neq t} Pr(w|NEG,t) \times \prod_{<w_i,w_j> \in MST} Pr\left(w_j|w_i,NEG,t\right)}\right) \\ &= \log\left(\prod_{w \in W, w \neq t} \frac{Pr(w|POS,t)}{Pr(w|NEG,t)}\right) + \log\left(\prod_{<w_i,w_j> \in MST} \frac{Pr\left(w_j|w_i,POS,t\right)}{Pr\left(w_j|w_i,NEG,t\right)}\right) \\ &= \sum_{w \in W, w \neq t} \log\left(\frac{Pr(w|POS,t)}{Pr(w|NEG,t)}\right) + \sum_{<w_i,w_j> \in MST} \log\left(\frac{Pr\left(w_j|w_i,POS,t\right)}{Pr\left(w_j|w_i,NEG,t\right)}\right) \\ &= f_U(s) + f_L(s) \end{aligned} \quad (7)$$

Finally, Eq. (7) represents the addition of two items, suggesting that the calculation of generation probability can be split into two parts. Obviously, the first part is a unigram model and the second part is a link model. We use f_U and f_L to refer to the decision functions based on unigrams and links, respectively. Such a combined form is close to the one in Nallapati's work (Nallapati and Allan, 2002). But their combined model for topic detection is just pieced together artificially, while ours is deducted from a general probability framework.

2.5. Sentence sentiment classification based on MST

If a MST is more likely to be generated by positive than by negative polarity, it has a higher chance of providing the positive perspective than the negative one, and vice versa. The generation probabilities derived from positive and negative views are regarded as the degrees to which the sensitive sentence approximates to the pos-

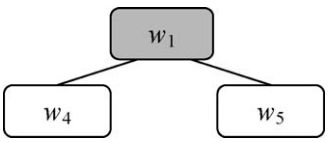

Fig. 5. A sub-tree instance from the MST with w1 as sub-tree root.

Algorithm 2: Sentiment Orientation Algorithm on MST for Sentence

Input: A MST which represents a sensitive sentence.

Output: the sentiment orientation (*SO*) of the sentence as the solution.

1: $SO = 0$;

2: for all $<t, w>$ in MST

3: $SO += \sum_w SO(w - subtree)$;

4: Output the *SO* of the sentence;

SO(*w-subtree*)

Input: A sub-tree with the root w in MST

Output: the sub-sentiment orientation (sub-*SO*) of current *w-subtree*.

5: $sub\text{-}SO = log\left(\frac{Pr(w \mid POS,t)}{Pr(w \mid NEG,t)}\right)$;

6: for all $<w, w_k>$ in *w-subtree*

7: $sub\text{-}SO = sub\text{-}SO + log\left(\frac{Pr(w_k \mid w,POS,t)}{Pr(w_k \mid w,NEG,t)}\right)$;

8: $sub\text{-}SO = sub\text{-}SO + SO(w_k\text{-}subtree)$;

9: Output *sub-SO*;

Fig. 6. Sentiment orientation algorithm on MST for a sensitive sentence.

itive or the negative orientation. Hence, the sentence classification algorithm based on the MST is illustrated in Fig. 6.

In this algorithm, we use *SO* (Sentiment Orientation) to indicate both the returned value and the name of function to get the sentiment orientation of a tree. Obviously, the called function $SO(w - subtree)$ is a recursive function which calculates the sentiment orientation of the current sub-tree. When the whole MST is recursively traversed, the sentiment orientation of the current sentence is returned. In the traversing procedure, the overall running time of this algorithm is $O(n)$. The "n" here is the number of the words in a sentence. Theoretically, in an MST, the links with weak contributions to express positive or negative opinion ought to have close generation probabilities in both positive and negative situations. They can be counteracted in the second part in Eq. (7).

2.6. Document sentiment classification

This study focuses on the sentiment classification of a document by comparing the generation probabilities of a collection of MSTs generated from the sensitive sentences (indicated by C^s) in the document. The difference is derived from the positive TTDMs (S^P) and the negative evaluation models (S^N).[1] The log-ratio decision function is defined in

[1] S^P is the set of positive TTDMs of all topical terms, and S^N has the corresponding meaning.

Eq. (8).

$$f(d) = log\left(\frac{Pr\left(C^s|S^P\right)}{Pr\left(C^s|S^N\right)}\right) = log\left(\frac{\prod_{MST \in C^s} Pr\left(MST|S^P\right)}{\prod_{MST \in C^s} Pr\left(MST|S^N\right)}\right) \tag{8}$$

Eq. (8) assumes the independence of the sensitive sentences in the document d. Take a sensitive sentence s_q in the document d for example. The symbols t_q, MST_q and W_q indicate the term contained in s_q, the corresponding MST structure and the word bag of s_q. Using the logarithm rule, Eq. (8) can be rewritten as Eq. (9).

$$\begin{aligned} f(d) &= \sum_{s_q \in C^s} \sum_{w \in W_q} log\left(\frac{Pr(w|POS, t_q)}{Pr(w|NEG, t_q)}\right) + \sum_{s_q \in C^s} \sum_{<w_i, w_j> \in MST_q} log\left(\frac{Pr(w_j|w_i, POS, t_q)}{Pr(w_j|w_i, NEG, t_q)}\right) \\ &= f_U(d) + f_L(d) \end{aligned} \tag{9}$$

By taking the combined form, it is convenient for us to compute each individual part in Eq. (9) one by one. Also the combined form allows us to balance the contributions of the components. As a result, the decision function can be converted to Eq. (10) by linear interpolation. We investigate the contribution of each part and the coefficient λ in Section 5.

$$f(d) = \lambda f_U(d) + (1-\lambda) f_L(d) : \begin{cases} > 0 & POS \\ < 0 & NEG \\ = 0 & NEUTRAL \end{cases} \tag{10}$$

Note that the neutral cases are not considered in this study because the documents in experiments are either tagged by "*POS*" or "*NEG*". On the other hand, it is reasonable to set two thresholds to identify the three situations instead of a simple threshold value "0". But in this study, we just use "0" as decision threshold.

Sentiment orientation of a document is the average orientation of its sensitive sentences in this study. The time complexity of this algorithm is $O(m)$, where "m" is the number of the sentences in the document. Therefore, the global time of classifying a document is $O(mn^2 log(n))$ when sentence words are considered as complexity analyzing unit, and "n" is the average word numbers of all sentences in document d.

3. Model parameter estimation

When identifying the opinion polarity of a document, two kinds of parameters, $Pr(w|Y, t)$ and $Pr(w_j|w_i, Y, t)$ in the TTDMs of topical term t, need to be estimated. Let all the sensitive sentences in training data consist of the training collection T_Y^s, i.e.

$$T_Y^s = \{s|Polarity(s) = Y\} \tag{11}$$

We assume the polarity of each training sensitive sentence is directly inherited from the training document that contains it. This makes it easy to compile training sentences since we do not need to annotate a large number of sensitive sentences manually and thus avoid a time-consuming effort. However, we are aware that a document with positive perspective may contain sentences that convey a negative point of view, and the opposite is also true. This issue will be addressed in our future work.

We use the maximum likelihood estimate (MLE) to estimate the parameters $Pr(w|Y, t)$ in the unigram models and the parameters $Pr(w_j|w_i, Y, t)$ in the link models. In addition, we choose the Kneser–Ney algorithm to smooth $Pr(w_j|w_i, Y, t)$.

3.1. MLE for $Pr(w|Y, t)$ and $Pr(w_j|w_i, Y, t)$

It has been concluded in Pang's study (Pang et al., 2002) that unigrams are credible in sentiment classification. Therefore, we simply employ MLE to estimate the unigram models but put more efforts on the link models with

additional consideration of the smoothing issue. That is,

$$Pr_{ML}(w|Y,t) = \frac{c(w|Y,t)}{c(*|Y,t)} \tag{12}$$

where $c(w|Y,t)$ is the number of times that the word w occurs in T_Y^s (Y indicates *POS* or *NEG*). $c(*|Y,t)$ is the total number of all words (indicated by "*") in T_Y^s. w and t appear in the same sentence, as well as $*$ and t.

Likewise, the MLE is applied to estimate the parameters of the link models.

$$Pr_{ML}(w_j|w_i,Y,t) = \frac{c(w_j|w_i,Y,t)}{c(*|w_i,Y,t)} \tag{13}$$

3.2. Kneser–Ney smoothing for link model

Kneser and Ney (1995) introduce an extension of absolute discounting smoothing by combining the lower order distribution with the higher order distribution. Chen and Goodman (1998) then advance the Kneser–Ney's algorithm by selecting the lower-order distribution. They demonstrate that Kneser–Ney smoothing performs best when compared with other commonly used smoothing techniques, by testing on different conditions. We apply Chen's revised Kneser–Ney algorithm here.

For the link models, we define the smoothed distribution Pr_{KN} to be the following modified form of Kneser–Ney smoothing.

$$Pr_{KN}(w_j|w_i,Y,t) = \frac{max\{c(w_i,w_j|Y,t)-D,0\}}{\sum_{w_j} c(w_i,w_j|Y,t)} + \frac{D}{\sum_{w_j} c(w_i,w_j|Y,t)} N_{1+}(w_i,\bullet|Y,t) Pr_{KN}(w_j|Y,t) \tag{14}$$

The notations N_{1+} is meant to evoke the number of words that have one or more counts, and the "$\bullet$" is meant to be a free variable that is summed over. We will write this value as $N_{1+}(w_i,\bullet|Y,t)$, formally defined as

$$N_{1+}(w_i,\bullet|Y,t) = |\{w_j : c\left(< w_i, w_j >\right) > 0\}| \tag{15}$$

In Eq. (14), D is the fixed discount from observed links and D equals $n_1/(n_1+2n_2)$ according to the suggestion from Ney. n_1 and n_2 are the total numbers of the bigrams with exactly one and two counts in the training data, respectively. In this paper, the original consecutive bigrams are modified to include the skipped bigrams i.e. word pairs, also. Specifically, we have Eq. (16) according to Chen and Goodman (1998).

$$Pr_{KN}(w_j|Y,t) = \frac{N_{1+}(\bullet,w_j|Y,t)}{N_{1+}(w_i,\bullet|Y,t)} \tag{16}$$

4. Experiment and discussion

The TTDMs are tested on two data sets. One is the corpus of "movie review" provided by Pang and Lee (2004). This corpus contains 1000 positive and 1000 negative reviews[2]. In all the experiments, only the stemmed words that occur more than twice in the 2000 reviews are considered. Stop words are excluded by our post-processing[3]. The remaining stemmed words constitute the vocabulary of the models. For more general details on this corpus, refer to Pang's papers.

The other one is the Chinese data set selecting product reviews harvested from the websites ZOL[4] and IT168.[5] All these reviews downloaded from the two websites are about electronic products, such as digital video cameras, mobile phones, and digital cameras. By the way, if a review consists of two sentiment parts: positive opinions and negative opinions, the two parts will be extracted as individual reviews with single sentiment orientations respectively.

Although the Chinese corpus consists of three topics, all the reviews are about digital products: digital video cameras, mobile phones, and digital cameras. The reviews from the three topics are very close in word usage and

[2] http://www.cs.cornell.edu/people/pabo/movie-review-data/.
[3] There are 294 words in our English stop-word list.
[4] http://www.zol.com.cn/.
[5] http://www.it168.com/.

Table 1
The contingency table of the statistics of the movie and digital product reviews.

Review genre	# of reviews	# of sensitive sentences
Movie		
Positive	1000	3381
Negative	1000	3333
Digital product		
Positive	730	2701
Negative	670	2680

Table 2
Contingency table.

	Tagged positive	Tagged negative
True positive	A	B
True negative	C	D

product description. In fact, we do not differentiate these three topics further and instead use the general label "Digital Product" in the following experiments.

To avoid domination of the corpus by a small number of prolific reviewers, the Chinese corpus imposes an upper limit of 20 reviews from one author and in one sentiment category. To identify whether a review (review 1) is redundant with respect to another review (review 2), we compare two reviews using the "dice coefficient":

$$dice = \frac{2 \times |W_1 \cap W_2|}{|W_1| + |W_2|} \tag{17}$$

In Eq. (17), W_1 or W_2 means the word bag of review 1 or review 2. If $d > 0.8$ here, the two reviews are marked as "similar". One of the two reviews will be removed from the review set. At last, we collected and labeled a corpus containing 730 positive and 670 negative reviews. Chinese reviews are pre-processed in a standard manner, i.e., they are segmented into words and Chinese stop words are removed.[6]

Because of the limitation of human resource, it is a little hard to collect a very large scale of non-redundant and useful Chinese corpus. But we still build the Chinese corpus progressively. The new data set will be used further.

Then, the Chinese sentence boundaries are found by identifying fixed Chinese characters. In total, 2701 and 2680 sentences containing topical terms are picked up from the positive and negative digital product reviews. As mentioned above, these sentences inherit the opinion polarity of the review they belong to. Table 1 shows the contingency table for the statistic of the movie and digital product reviews respectively.

To measure the reliability of the polarity inheritance, we conducted an agreement statistic, randomly picking 70 reviews that contain 271 sensitive sentences for English movie reviews and 67 reviews containing 258 sensitive sentences for Chinese digital product reviews. The overall portion of correct inheritance of sentiment polarity is 0.91 for movie reviews and 0.96 for digital product reviews. The inheritance precisions are considered to be enough for model learning.

The labeled documents are randomly divided into 10 collections, then we can perform 10-fold cross validation. In test processes, a document may be classified into positive or negative. That is to say, there exist two kinds of classification errors called "false negative" and "false positive". Thus, we build Table 2 for evaluation purpose.

In the table, A, B, C and D indicate the number of each case. When the system classifies a true positive document into "positive" or classifies a true negative document into "negative", these two are correct, yet the other two cases are wrong. Therefore, the accuracy is defined as a global evaluation mechanism:

$$Accuracy = \frac{(A + D)}{(A + D) + (B + C)} \tag{18}$$

[6] There are 177 words in our Chinese stop-word list.

Table 3
The top 30 topical terms in English movie reviews.

"film", "movie", "character", "scene", "time", "story", "play", "plot", "show", "performance", "star", "actor", "director", "action", "role", "audience", 'comedy", "fact", "cast", "script", "act", "part", "screen", "picture", 'hollywood", "feature", "series", "writer", "dialog", "box"

Table 4
The top 37 topical terms(translated from Chinese) in Chinese digital product reviews.

"color", "resolution", "memory card", "camera", "fitting", "tripod", "battery", "memory size", "interface", "appearance", "earphone", "exposal", "frame-rate", "auto-focus", "accessory", "screen", "function", "flashlight", "size", "Signal-to-Noise", "pixel", "lens", "slot", "power", "machine type", "lightness", "contrast", "player", "keyboard", "button", "game", "after-sale fault rate", "tone", "cost-effective ratio", "software", "data transmission", "body"

Obviously, the larger the accuracy, the better the system performance. The 10-fold cross validation based average accuracy is the major evaluation measure in the following experiments.

4.1. Experiments on topical term extraction

Since there is no difference between the experiments on both the corpora after the Chinese text is segmented into words, we use English movie reviews as example for the description of topical term extraction.

In this study, the topical terms are semi-automatically extracted from the movie corpus (M) by comparing M with a non-movie review background corpus (B)[7], and then selecting the top-ranked terms as the topical terms manually.

Intuitively, the topical terms ought to be domain relevant and rarely appear in other domains. We investigate the Inverse Word Frequencies (IWF), defined in Basili et al. (1999) as the filter (i.e. the logarithm part in Eq. (19)). Given a candidate w, assume its frequency in the movie review corpus is $fre_M(w)$ and its frequency in the background corpus is $fre_B(w)$. The termhood of w is defined by Eq. (19).

$$Termhood\,(w) = fre_M\,(w) \log\left(\frac{N}{fre_B\,(w)}\right) \tag{19}$$

where N is the size of the background corpus. Human selection is intervened after termhood calculation. At last, we select the top 30 words ranked by termhood as the topical terms (see Table 3) for English movie reviews. The same procedure is applied to extract the Chinese topical terms from digital product reviews. At last, we select the top 37 topical terms (see Table 4) for Chinese digital product reviews. Although a variety of methods have been proposed to select the most significant domain terms, the focus of our work is to examine the effect of the TTDMs. We do not make further efforts to extract topical terms.

Note that topical term extraction is a semi-automatic process. It relies on human selection once the candidate terms have been ranked by termhood.

4.2. Experiments on sentiment classification

Three sets of experiments are conducted to evaluate and discuss the proposed approach.

1. The first set of experiments changes the value of λ in the combined model to examine how the link models contribute to the combined models.
2. The second set of experiments checks if the number of the topical terms is a key feature.

[7] We use a generic corpus (reuters21578) as a background corpus for English and Trec6's XinHua corpus for Chinese.

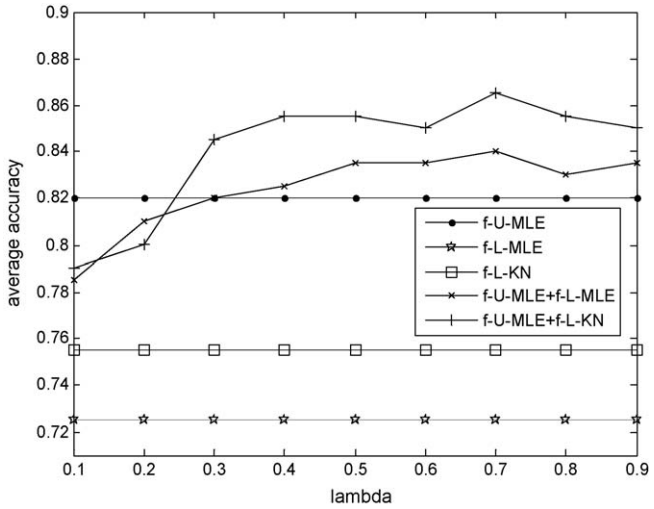

Fig. 7. Average accuracy by tuning x value of λ for English movie reviews.

3. Finally, the last set of experiments compares five different models by setting $\lambda = 0.7$ for English movie reviews and $\lambda = 0.8$ for Chinese digital product reviews. We use the top 10 topical terms for English movie reviews and the top 23 topical terms for Chinese digital product reviews, to guide the link extraction.

We present five different models to compare results. They are the unigram model (i.e. f_{U_MLE}), the link only model estimated by MLE (i.e. f_{L_MLE}), the smoothed link only model estimated by Kneser–Ney smoothing (i.e. f_{L_KN}), the non-smoothed combined model (i.e. $f_{U\_MLE} + f_{L\_MLE}$) and the smoothed combined model (i.e. $f_{U\_MLE} + f_{L\_KN}$). Note that all these five models are discussed in the three sets of experiments.

If we do not know what is the best lambda and the suitable number of topical terms for movie reviews and digital product reviews respectively in advance, we have to use a series of experiments to find the best value of these parameters. Fig. 7 then plots the average accuracy of a series of comparative experiments by tuning the value of the parameter λ from 0.1 to 0.9 with the step of 0.1 for English movie reviews. And similarly, Fig. 8 tunes the value of the parameter λ from 0.1 to 0.9 with the step of 0.1 for Chinese digital product reviews. These experiments are conducted for all the five different models presented. To both English and Chinese test sets, the smoothed combined model stands above the non-smoothed combined model at most λ value points benefits from the use of Kneser–Ney smoothing. Besides,

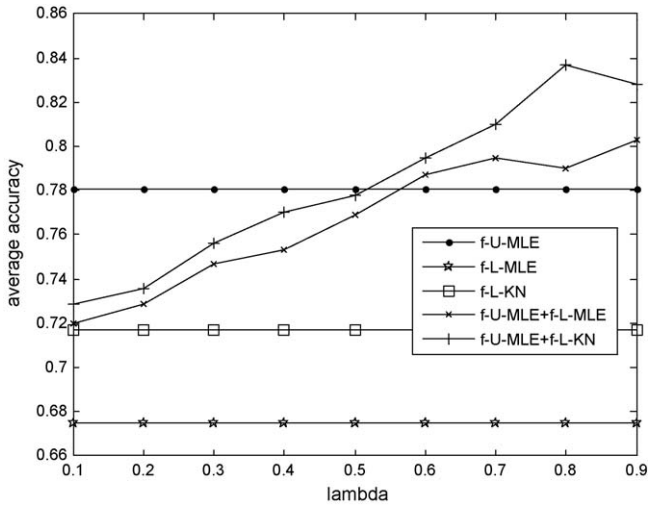

Fig. 8. Average accuracy by tuning x value of λ for Chinese digital product reviews.

Table 5
Average accuracies by choosing different number of topical terms for English.

# of terms	2	4	6	8	10
Avg. acc.	0.815	0.830	0.830	0.845	**0.865**
# of terms	12	15	20	25	30
Avg. acc.	0.840	0.845	0.835	0.820	0.830

it is easy to conclude from Figs. 7 and 8 that the links make incontestable contribution to the overall performance of the combined models for most value points. When λ equals 0.7 for movie reviews and λ equals 0.8 for digital product reviews, it achieves the best result for model $f_{U\_MLE} + f_{L\_KN}$.

In Eq. (10), λ tunes the importance of f_U and f_L. The larger λ is, the more contributions f_U makes. Both Figs. 7 and 8 illustrate that it makes sense to use the link model as an effective factor for sentiment classification, but the unigram model makes more contribution. The small training size might explain why the unigram model is a more important factor. The data sparseness derived from small training size might also explain why both the sentiment classification of movie and digital product reviews depend more on the unigram model. It is harder to mine very powerful statistical information for links, thus the link model contributes less to sentiment classification than the unigram model.

On the other hand, we also try the Expectation Maximization (EM) (Dempster et al., 1977) algorithm to find suitable settings of λ. We estimate λ by maximizing the log-likelihood of all training instances, given the interpolation model:

$$\lambda = \underset{\lambda}{argmax} \sum_{k=1}^{|\overset{s*}{T}_Y|} \sum_{<w_i, w_j> \in MST_k} log(\lambda Pr(\overset{(k)}{w}_i | Y, t^{(k)}) + (1-\lambda) Pr(\overset{(k)}{w}_j | \overset{(k)}{w}_i, Y, t^{(k)})) \tag{20}$$

λ can be estimated using the EM iteration procedure:

1. Initialize λ to a random estimate between 0 and 1.
2. Update λ using:

$$\lambda' = \frac{1}{|T_Y^s|} \times \sum_{k=1}^{|\overset{s}{T}_Y|} \sum_{<w_i, w_j> \in MST_k} \frac{(1-\lambda) Pr(\overset{(k)}{w}_j | \overset{(k)}{w}_i, Y, t^{(k)})}{\lambda Pr(\overset{(k)}{w}_i | Y, t^{(k)}) + (1-\lambda) Pr(\overset{(k)}{w}_j | \overset{(k)}{w}_i, Y, t^{(k)})} \tag{21}$$

3. Repeat Step 2 until λ converges.

Where T_Y^s denotes all training instances with polarity "Y", and $|T_Y^s|$ is the number of training sentences which is used as a normalization factor. We set λ to 0.1 according to the experimental results. The final iterative value of λ is 0.644 and 0.870 for movie reviews and digital product reviews, respectively, both of them are close to the manually tuned value of λ (0.7 and 0.8) illustrated in Figs. 7 and 8.

To see whether more terms can provide more informative links, we conduct another set of experiments by choosing the number of topical terms from 2 to 30 for movie reviews and the number of topical terms from 3 to 37 for digital product reviews. These experiments are conducted with the smoothed combined model and the parameter λ is set to 0.7 and 0.8, respectively. The results in Tables 5 and 6 suggest that 10 and 23 are good choices for movie reviews and digital product reviews. The conclusion is that it is not necessary to use a larger term set for the current data set.

Though the topical terms are semi-automatically selected, i.e., we have removed the obviously non-topical terms, some terms are still "noisy" to sentiment classification. The main reason might lie in the limitation of data. The TTMDs of the less frequent topical terms could not be learnt sufficiently from the training data. This issue happened to both the movie and digital product reviews. It takes more efforts and time to supply the reviews data set. In our work, there exist "noisy" terms which are topical terms in fact. The better topical terms in the two corpora are listed in Tables 7 and 8, respectively.

Table 6
Average accuracies by choosing different number of topical terms for Chinese digital product reviews.

# of terms	3	6	9	12	15
Avg. acc.	0.747	0.791	0.783	0.797	0.805
# of terms	18	23	27	32	37
Avg. acc.	0.809	**0.837**	0.828	0.801	0.785

Table 7
Better topical terms for movie reviews.

"film", "movie", "character", "scene", "time", "story", "plot", "director", "fact", "script"

Table 8
Better topical terms (all are written in Chinese) for digital product reviews.

"color", "resolution", "battery", "memory", "interface", "appearance", "accessory", "screen", "function", "flashlight", "size", "pixel", "lens", "power", "contrast", "player", "button", "after-sale service", "tone", "cost-effective ratio", "software", "data transmission", "body"

The third set of experiments show the model performances according to the average accuracy by 10-fold cross validation. When selecting or tuning parameters and doing cross validation, it is better to divide the data into training, development, and testing sets, with the development set being used to select/tune parameters for just that fold. See Raaijmakers's analysis (Raaijmakers, 2007). Because of the limitation of the data size, it is a little hard to prepare a special data set for development use. So we only divide the data into training and testing sets in 10-fold cross validation.

The second column in Tables 9 and 10 presents the results of the five models. The next two columns are the percentages of the changes over the unigram model and the link model estimated by MLE only, respectively.

It shows that the smoothed combined model consistently performs the best and achieves an improvement of 5.5% and 7.1% when compared with the unigram model for the two corpora, respectively. That is to say, the improvement

Table 9
Average accuracy of five models on English movie reviews.

Models	Avg. acc.	Change (%) over f_{U_MLE}	Change (%) over f_{L_MLE}
f_{U_MLE}	0.820	–	+13.1 *
f_{L_MLE}	0.725	−11.6 *	–
f_{L_KN}	0.755	−7.3 *	+4.1
$f_{U\_MLE} + f_{L\_MLE}$	0.840	+2.5	+15.9 *
$f_{U\_MLE} + f_{L\_KN}$	0.865	+5.5 *	+19.3 *

* The difference is statistically significant according t-test ($p < 0.01$).

Table 10
Average accuracy of five models on Chinese digital product reviews.

Models	Avg. acc.	Change (%) over f_{U_MLE}	Change (%) over f_{L_MLE}
f_{U_MLE}	0.781	–	+15.7 *
f_{L_MLE}	0.675	−20.0 *	–
f_{L_KN}	0.717	−8.2 *	+6.2 *
$f_{U\_MLE} + f_{L\_MLE}$	0.803	+2.8	+18.9 *
$f_{U\_MLE} + f_{L\_KN}$	0.837	+7.1 *	+24.0 *

* The difference is statistically significant according t-test ($p < 0.01$).

by integrating the links is promising. In addition, the smoothed link models outperform the non-smoothed link models, either combined with the unigram model or not.

Intuitively, links ought to contain more classification information than unigrams do. However, a link model has a larger parameter space than corresponding unigram model, and it is hard to search the best parameter set for link models on the limited training data. Thus, it is understandable that the link models alone clearly perform the worst now. That is why currently the unigram models (smoothed or not) are still more important in the combined models.

The experiment also shows that the improvement is more pronounced on Chinese digital product reviews than English movie reviews, that is to say, the link model contributes more to the Chinese corpus. It is meaningless to compare the movie reviews in English and digital products reviews in Chinese since they share little common ground. But, the experiments show the general trend that the link model will improve the performance of the polarity classifier. The reason that the link model contributes more to the Chinese reviews here is that the description of digital products in any two product reviews are closer than in any two movie reviews. Since the human resource is limited, and a corpus of high quality Chinese digital product reviews is hard to assemble, the problem of data sparseness for Chinese sentiment classification (as well as the English movie reviews) can not be satisfactorily solved unless a good corpus is available.

Overall, the results are encouraging. The accuracy of the smoothed combined model for the English movie reviews is comparable to the published results which are around 0.86 on this data set and by the 10-fold cross validation (Pang and Lee, 2004). The result also achieves the state of art of sentiment classification on this raw corpus.[8] However, there is still much room for improvement. Both the experiments show that link models alone can hardly handle the data sparseness problem. The functions of link models have to be fully put into practice by combining other stronger models.

5. Related work

As one of the opinion mining tasks, sentiment classification has attracted tremendous research attention for its broad applications in many domains, such as movie reviews and customer feedback reviews.

Sentiment classification can be performed on words, sentences, documents, or sentences and documents simultaneously. Our work is a novel method that focuses on individual words, i.e., topical terms before performing the document sentiment classification.

In terms of identifying semantic orientation of individual words or phrases, a large body of research focuses on this task by employing linguistic heuristics (e.g., Hatzivassiloglou and McKeown, 1997; Kim and Hovy, 2004; Turney, 2002; Turney and Littman, 2003). Often, a measure of the strength of sentiment polarity is developed to determine how strongly a word or phrase is judged positive or negative. Hatzivassiloglou and McKeown (1997) study the effects of adjectives only. They argue that the adjectives are strong predictors. However a lot of words/phrases with other parts-of-speech can also express sentiment information. Kim and Hovy (2004) developed two models for word sentiment classification. The basic approach is to assemble a small amount of positive and negative seed words by hand, and then to grow this by adding words obtained from WordNet. They assume synonyms of positive words are mostly positive and antonyms mostly negative, and vice verse. Turney (2002) and Turney and Littman (2003) determine the semantic orientation of a phrase by computing its Pointwise Mutual Information (PMI) value with pre-defined seed words, such as “excellent” and “poor”. Then the sentiment information of the phrases are averaged to predict the sentiment orientation of a review. Our study is close to the idea of identifying opinion polarity of single words first, then the whole document. The words in our work are nothing more than topical terms.

On the other hand, the attention is fixed on identifying the opinion of whole documents at first. With respect to this, machine learning approaches become very predominant, they are also applied to recognizing sentiment polarity. By comparing different learning algorithms, Pang et al. (2002) conclude that SVM in general obtains better results with unigram features. Later on, Pang and Lee (2004) further advance their previous study by training a sentence subjectivity classifier and then determining document sentiment polarity relying on the identified sensitive sentences. Sentiment classification is more like a pattern recognition task in these efforts. In the work of Bai et al. (2004), Bayesian belief networks are used to represent a Markov Blanket, i.e. a directed acyclic graph on which each vertex; represents a word and the edge corresponds to the parent/child relationship between the words. They find that a vocabulary is efficient for

[8] http://www.cs.cornell.edu/people/pabo/movie-review-data/otherexperiments.html.

the purpose of extracting sentiments. But it is hard to use linguistic properties in their work because the links between words are mined from a document.

Very recently, the mutual influence of sentence-level and document-level sentiment polarities are stressed by McDonald et al. (2007). They investigate a joint sentiment classification framework which incorporates the classifications at both the sentence and document levels. As a matter of fact, numerous research articles focusing on document classification have utilized sentence analysis. In Pang's work (Pang and Lee, 2004), an initial model identifies subjective sentences that are considered as containing more sentiment information. The top subjective sentences are then input into a standard document level polarity classifier with improved results. Isotonic (2006) used a sequential isotonic CRF model to measure polarity of sentence in order to determine the sentiment flow of authors in documents. McDonald et al. (2007) investigate a structured model for jointly classifying the sentiment of text at varying levels of granularity. The model learns to predict sentiment on the sentence level of granularity for a document by CRF and finds that the sentence information is advantageous to document classification. Our work differs from these papers by proposing a relatively complex sentiment structure of a topical term in a corresponding sentence. And then the document level analysis can be achieved.

To date, several studies have been performed to use rich English resources for sentiment classification in other languages. Standard Naive Bayes and SVM classifiers have been applied for subjectivity classification in Romanian (e.g., Banea et al., 2008; Mihalcea et al., 2007), and the results show that automatic translation is a viable alternative for the construction of resources and tools for subjectivity analysis in a new target language. Wan (2009) applies co-training method based on machine translation to making use of unlabeled data for Chinese review sentiment classification.

Our method can also be categorized as an effort of sentiment mining , which focuses on mining the sentence contextual structure of the topical terms to determine the sentiment orientation, while some efforts of mining opinions from sentences have been made on the task of extracting such templates as $< who$ feels how about $whichaspect$ of $whatproduct >$ from unstructured text data. For example, Kobayashi et al. (2007) conduct research on the extraction of "aspect-evaluation" and "aspect-of" relations within sentences. They propose a machine learning-based method which combines syntactic patterns and statistical clues. But these relations are not used for determining the sentiment orientation of a document. In contrast to sentiment classification, opinion extraction aims to produce more discrete information and requires an in-depth analysis of opinions (e.g., Hu and Liu, 2004; Ku et al., 2006; Popescu and Etzioni, 2005; Yi et al., 2003). For instance, Hu and Liu produce opinion summaries of products, categorized by the opinion polarity, while Ku et al. (2006) propose algorithms for opinion extraction at word, sentence and document levels, and then summarize topical and opinionated information. Obviously, mining all sentiment about slots is a more complex task.

Besides the classification and mining of sentiment, another growing interest is the ranking of sentiment in one topic retrieval process. Representative works are credited to Eguchi and Lavrenko (2006) who consider topic dependence of sentiment. The topic and sentiment classification are performed at the same time in a unified sentiment information retrieval model according to both the topic and the sentiment relevance to a user's information need. Their model is developed in the framework of generative modeling approach and works significantly better than the standard language modeling retrieval approaches that do not have special treatment for sentiment.

Pang and Lee (2008) gives a survey to cover techniques and approaches that promise to directly enable opinion-oriented information-seeking systems, and to convey to the reader a sense of excitements about the intellectual richness and breadth of the area.

6. Conclusion

In this paper, we present a novel topical term description modeling approach to sentiment classification.

1. We assume that a topical term and its context can help to determine the sentence polarity;
2. We investigate the ability of capturing sentiment information by constructing MSTs;
3. We create sentiment TTDMs by learning from the topical terms and their specified contexts;
4. We also discuss how to refine the TTDMs through some smoothing techniques.

The experiments on the publicly available movie review corpus and digital product corpus show that the proposed approach to sentiment classification is reasonably effective. In particular, the improvement from the TTDMs is encour-

aging. It shows that the proposed approach is able to learn the positive and negative contextual knowledge effectively in a supervised manner.

This study may suggest a direction to develop more effective TTDMs for sentiment classification. It drives us to further study how to define and make use of the link information appropriately and effectively.

It should be noted that our goal is not to show that our approach can perform much better than the classical machine learning methods, but to investigate the role of TTDMs in sentence-document level sentiment classification. In the future, we plan to improve the proposed model based method by exploring some hierarchical link models, and integrating conceptual features into these hierarchical models.

It is worth considering a dependency tree based representation for sentence. A dependency tree is also a powerful structure for representing a sentence, which is good at analyzing the syntactic structure in the sentence, but some links for sentiment might be weakened since the words are far in a dependency tree. How to use dependency information for sentiment classification will also be considered in the future work.

Another problem in this paper is that a sensitive sentence is supposed to express opinions to the topical term. But it is best to exactly decide the subjectivity of the sentence in advance for classification in the next step, as sentence subjectivity analysis is not a trivial task (Pang and Lee, 2004). Also, regarding the fact that not all sentences in a review are sensitive, it is an interesting research in the future to tell the subjectivity of a sentence containing a topical term.

We are also aware that a document with positive perspectives may contain sentences that convey a negative point of view, and vice versa. Strictly annotating a training corpus is also very useful for future research.

Acknowledgements

The work presented in this paper is supported by the Research Grants Council of Hong Kong (Project Numbers: CERG PolyU5211/05E and CERG PolyU5230/08E) and the Hong Kong Polytechnic University Internal Grant (Account No. G-YH53).

Appendix A.

Table A.1 below shows the sample links directly associating the terms “film” with the links “¡well, slip¿”, “¡crazy, thing¿” and “¡art, best¿” in English movie reviews. “–” means the corresponding link does not occur. Table A.2 shows the examples of the links directly associating the terms “battery (in Chinese)”, “¡power (in Chinese), strong (in Chinese)¿” and “¡capacity (in Chinese), not enough (in Chinese)¿” and “¡market (in Chinese), praise(in Chinese)¿” in Chinese digital product reviews. “–” means the corresponding link does not occur.

The values in the tables denote the probabilities of these links in either the positive or the negative models, which are true in accordance to our general understanding.

Table A.1
Generation probabilities of example links from English movie reviews.

	Film		
	<well, slips>	<crazy, thing>	<art, best>
Positive	0.001838	0.001003	0.002875
Negative	0.000312	0.004471	–

Table A.2
Generation probabilities of example links from Chinese digital product reviews.

	Battery (in Chinese)		
	<power (in Chinese), strong (in Chinese)>	<capacity (in Chinese), not enough (in Chinese)>	<market (in Chinese), praise (in Chinese)>
Positive	0.037120	0.002033	0.003675
Negative	–	0.005072	–

References

Bai, X., Padman, R., Airoldi, E., 2004. Sentiment extraction from unstructured text using tabu search-enhanced markov blanket. In: Proceedings of International Workshop on Mining for and from the Semantic Web, 2004.

Banea, C., Mihalcea, R., Wiebe, J., Hassan, S., 2008. Multilingual subjectivity analysis using machine translation. In: Proceedings of EMNLP.

Basili, R., Moschitti, A., Pazienza, M., 1999. A text classifier based on linguistic processing. In: Proceedings of IJCAI, Machine Learning for Information Filtering.

Chen, S.F., Goodman, J., 1998. An empirical study of smoothing techniques for language modeling. Technical Report: TR-10-98, Harvard University.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society 39, 1–38.

Eguchi, K., Lavrenko, V., 2006. Sentiment retrieval using generative models. In: Proceedings of EMNLP, pp. 345–354.

Hatzivassiloglou, V., McKeown, K., 1997. Predicting the semantic orientation of adjectives. In: Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL, pp. 174–181.

Hu, M., Liu, B., 2004. Mining and summarizing customer reviews. In: Proceedings of the 10th International Conference on KDD, pp. 168–177.

Isotonic conditional random fields and local sentiment flow. In: Proceedings of NIPS, 2006, pp. 961–968.

Kennedy, A., Inkpen, D., 2006. Sentiment classification of movie and product reviews using contextual valence shifters. Computational Intelligence 22 (2), 110–125.

Kim, S., Hovy, E., 2004. Determining the sentiment of opinions. In: Proceedings of the 20th International Conference on Computational Linguistics, pp. 1367–1373.

Kneser, R., Ney, H., 1995. Improved backing-off for m-gram language modeling. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1.

Kobayashi, N., Inui, K., Matsumoto, Y., 2007. Extracting aspect-evaluation and aspect-of relations in opinion mining. In: Proceedings of EMNLP, pp. 1065–C1074.

Ku, L., Liang, Y., Chen, H., 2006. Opinion extraction, summarization and tracking in news and blog corpora. In: Proceedings of the 21st AAAI, pp. 100–107.

Manning, C., Schutze, H., 1999. Foundations of Statistical Natural Language Processing. MIT Press, pp. 181–184.

McDonald, R., et al., 2007. Structured models for fine-to-coarse sentiment analysis. In: Proceedings of the 45th ACL, pp. 432–439.

Mihalcea, R., Banea, C., Wiebe, J., 2007. Learning multilingual subjective language via cross-lingual projections. In: Proceedings of the 45th ACL.

Nallapati, R., Allan, J., 2002. Capturing term dependencies using a language model based on sentence trees. In: Proceedings of the 11th CIKM, pp. 383–390.

Pang, B., Lee, L., 2004. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd ACL, pp. 271–278.

Pang, B., et al., 2002. Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of EMNLP.

Pang, B., Lee, L., 2008. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2 (1–2), 1–135.

Popescu, A., Etzioni, O., 2005. Extracting product features and opinions from reviews. In: Proceedings of EMNLP, pp. 339–346.

Raaijmakers, S., 2007. Sentiment classification with interpolated information diffusion kernels. In: Proceedings of the 1st International Workshop on Data Mining and Audience Intelligence for Advertising, pp. 34–39.

Rigsbergen, V., 1979. Information Retrieval. Butterworths.

Turney, P.D., 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of ACL, pp. 417–424.

Turney, P.D., Littman, M.L., 2003. Measuring praise and criticism: inference of semantic orientation from association. ACM Transactions on Information Systems (TOIS) 21 (4), 315–346.

Wan, X., 2009. Co-training for cross-lingual sentiment classification. In: Proceedings of the 47th ACL and the 4th IJCNLP of the AFNLP, pp. 235–243.

Yi, J., et al., 2003. Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques. In: Proceedings of the 3rd ICDM, pp. 427–434.